MARLIN: Masked Autoencoder for facial video Representation LearnING

Zhixi Cai¹, Shreya Ghosh¹,², Kalin Stefanov¹, Abhinav Dhall¹,³, Jianfei Cai¹, Hamid Rezatofighi³, Reza Haffari¹, Munawar Hayat¹
¹Monash University, ²Curtin University, ³Indian Institute of Technology Ropar
{zhixi.cai,kalin.stefanov,jianfei.cai,hamid.rezatofighi,gholamreza.haffari, munawar.hayat}@monash.edu, shreya.ghosh@curtin.edu.au, abhinav@iitrpr.ac.in

Abstract

This paper proposes a self-supervised approach to learn universal facial representations from videos, that can transfer across a variety of facial analysis tasks such as Facial Attribute Recognition (FAR), Facial Expression Recognition (FER), DeepFake Detection (DFD), and Lip Synchronization (LS). Our proposed framework, named MARLIN, is a video masked autoencoder that learns highly robust and generic facial embeddings from abundantly available non-annotated web crawled facial videos. As a challenging auxiliary task, MARLIN reconstructs the spatio-temporal details of the face from the densely masked facial regions which mainly include eyes, nose, mouth, lips, and skin to capture local and global aspects that in turn help in encoding generic and transferable features. Through a variety of experiments on diverse downstream tasks, we demonstrate MARLIN to be an excellent facial video encoder as well as feature extractor, that performs consistently well across a variety of downstream tasks including FAR (1.13% gain over supervised benchmark), FER (2.64% gain over unsupervised benchmark), DFD (1.86% gain over unsupervised benchmark), LS (29.36% gain for Frechet Inception Distance), and even in low data regime. Our code and models are available at https://github.com/ControlNet/MARLIN.

1. Introduction

Facial analysis tasks [34, 43, 70, 85] provide essential cues for human non-verbal behavior analysis, and help unfold meaningful insights regarding social interaction [36], communication [40], cognition [68] with potential applications in Human-Computer Interaction (HCI) and Affective Computing domains. Recently, we have witnessed significant progress in deep neural network models to solve facial analysis tasks such as Facial Attribute Recognition (FAR) [34,85], Facial Expression Recognition (FER) [48], DeepFake Detection (DFD) [70], and Lip Synchronization (LS) [43]. While these deep models can achieve remarkable performance, they often require large-scale annotated datasets, which is not only a resource-expensive and time-consuming process but also infeasible for some applications requiring domain expertise for annotation (e.g. FER).

To this end, self-supervised pre-training [26, 37, 71] has lately emerged as an effective strategy to address the limitations of fully supervised methods, as it enables generic representation learning from non-annotated data, that can then be transferred across tasks having limited labels. For images of natural scenes and objects, self-supervised learning approaches using self-distillation [14], contrastive-learning [19,18], solving pre-text tasks such as jigsaw puzzle [53], and more recently autoencoding [37,71] have even outperformed the supervised learning approaches.

Despite the promises offered by these self-supervised methods in learning scalable and generic representations for natural scene images and videos, these have not yet been investigated for learning representations from facial video data. Facial representation learning requires tracking of fine-grained face specific details which might not be perfectly captured by linear tube masking [71]. Until now, most of the existing approaches associated with facial analysis tasks are highly specialized and develop...
task-specific models trained in a fully supervised manner [46, 54, 63], with very few recent efforts towards learning generic image-based facial encoding [10, 84]. These closely related works [10, 84] either focus on exploring training dataset properties in terms of size and quality [10] or performing pre-training in visual-linguistic way [84]. These works [10, 84] are hard to scale since they use static image-level facial information and the image-caption pairs are highly associated with context information rather than face.

In this paper, our goal is to learn universal and task-agnostic representations in a self-supervised manner for face-related downstream tasks (see Fig. 1). For this purpose, we employ a masked autoencoder [37, 71] with a facial-guided masking strategy that learns to reconstruct spatio-temporal details of a face from densely masked facial regions using non-annotated videos. Unlike existing approaches for natural scene videos [71], where the tube-masking is initialized with a static part of the video without any semantic information, our approach dynamically tracks face and then develops a facial part-guided tube masking strategy using an off-the-shelf face parser i.e. FaceX-Zoo [75]. Thus, we pose a more challenging task that encourages the model to learn spatio-temporal representations to cover local as well as global information. Inspired by prior works [27, 60] showing high-quality reconstruction results along with rich and generic latent features, we incorporate adversarial loss on top of masked encoding to enhance reconstruction quality. Our experimental results show that our proposed framework, MARLIN, learns highly generic facial encoding that scale and transfers well across diverse facial analysis tasks such as FER, DFD, FAR, and LS and achieve favorable performance gain w.r.t. state-of-the-art benchmarks. In summary, our main contributions are:

- We propose, MARLIN, a universal and task-agnostic facial encoder that learns robust and transferable facial representation from abundantly available non-annotated web-crawled facial videos in a self-supervised fashion.
- As a challenging auxiliary task, we propose to reconstruct the spatio-temporal details of the face from the densely masked facial regions. The proposed facial region-guided tube masking (aka Fasking) strategy aims to learn local and global aspects from facial videos which in turn help encode generic and transferable features.
- Through extensive quantitative and qualitative analysis, we show that MARLIN learns rich, generic, transferable, and robust facial representation, that performs consistently well across a variety of downstream tasks including FAR (1.13% gain over supervised benchmark), FER (2.64% gain over unsupervised benchmark), DFD (1.86% gain over unsupervised benchmark), LS (29.36% gain for Frechet Inception Distance) and even in few shot settings.

### 2. Related Work

**Masked AutoEncoder.** Masked autoencoder learns robust and transferable representations based on the hypothesis of reconstruction of the masked region. Masked autoencoding is motivated by context encoders [56] and denoising encoders [73]. After success of BERT [26] based masking, the vision community has also explored different design choices of masked auto encoding such as pixel level masking [17, 37, 80], token level masking [29] and deep feature based masking [6, 77], using vision Transformers [44, 52]. Similarly, for modeling spatio-temporal patterns of the input data, masked motion modelling [69] and tube masking [71] strategies have been incorporated recently. Along this line, MARLIN masks and reconstructs domain-specific facial parts to learn universal facial representation.

**Facial Representation Learning.** Till date, most of the existing facial analysis approaches are conducted in a task-specific way with fully supervised manner [46, 54, 63] on manually annotated data to enhance performance. Any state-of-the-art model’s performance on benchmarked datasets is impacted by the quality and quantity of annotated data used during training. Tab. 1 shows an overview of the task-specific large-scale facial image or video datasets that have been curated over the past decade [1] to facilitate research in Face Verification (LFW [39], MS-celeb-1M [34], VGG-FACE [54], VGGFace2 [13]), Facial Attribute Recognition(CelebA [50], CelebV-HQ [85]), Facial Emotion Recognition (CMU-MOSEI [83]), DeepFake Detection (FF++ [62]) and Lip Synchronization (LRS2 [22]). However, data curation encounters several challenges such as requirements of specialized hardware (e.g. for FER and action unit data), the discrepancy in data distribution that leads to prevent merging of multiple datasets [10], and most importantly time consuming and resource expensive annotation process. To eliminate these drawbacks, some of the existing approaches [20, 81, 82] adopt data augmentation strategy via image or video synthesis as the surge in face generation technology fueled by Generative Adversarial Network (GAN) [20, 67, 81, 82] and other generation techniques [16, 55] aids realistic face generation even with the

<table>
<thead>
<tr>
<th>Datasets</th>
<th># Samples</th>
<th>Env.</th>
<th>Fmt.</th>
<th>Task</th>
<th>Year</th>
</tr>
</thead>
<tbody>
<tr>
<td>VGG-FACE [54]</td>
<td>2.6M</td>
<td>Wild</td>
<td>Img.</td>
<td>Identification</td>
<td>2015</td>
</tr>
<tr>
<td>YouTubeFace [78]</td>
<td>3,425</td>
<td>Wild</td>
<td>Vid</td>
<td>Identification</td>
<td>2011</td>
</tr>
<tr>
<td>LRS2 [22]</td>
<td>144,482</td>
<td>Wild</td>
<td>Vid</td>
<td>Lip Sync.</td>
<td>2017</td>
</tr>
<tr>
<td>CelebV [79]</td>
<td>5</td>
<td>Wild</td>
<td>Vid</td>
<td>Reenact</td>
<td>2018</td>
</tr>
<tr>
<td>CMU-MOSEI [83]</td>
<td>23,453</td>
<td>Wild</td>
<td>Vid</td>
<td>Emo, Senti</td>
<td>2018</td>
</tr>
<tr>
<td>FaceForensics++ [62]</td>
<td>1,004</td>
<td>Wild</td>
<td>Vid</td>
<td>DeepFake</td>
<td>2019</td>
</tr>
<tr>
<td>VoxCeleb2 [23]</td>
<td>150,480</td>
<td>Wild</td>
<td>Vid</td>
<td>Speaker</td>
<td>2018</td>
</tr>
<tr>
<td>Celeb-V-HQ [85]</td>
<td>55,666</td>
<td>Wild</td>
<td>Vid</td>
<td>Attribute</td>
<td>2022</td>
</tr>
</tbody>
</table>

**Table 1. Facial Analysis Tasks.** Overview of different face related tasks and relevant datasets down the lane.
control over facial attributes. These generation techniques add variation in training set quantitatively, but in some cases it still lags in qualitative aspects due to domain specific inconsistency and more importantly high network complexity.

To this end, there are very few recent works that aim to learn image-based task specific facial encoding with limited supervision [3, 9, 10, 65, 84, 86, 88]. The most closely related existing works [10, 84] either focus on exploring training dataset properties in terms of size and quality [10] or performing pre-training in visual-linguistic way [84]. These works [10, 84] are hard to scale since they use static image level facial information and the image-caption pairs are highly correlated with context information rather than face. In this work, we aim to develop a generic, universal, and task-agnostic facial encoder that learns from web-crawled non-annotated data. Our experimental analysis shows that MARLIN can align the latent space manifold to any desired downstream task specific label space. Thus, MARLIN has the capability to act as a strong facial encoder or feature extractor in many low-resource real world applications.

3. MARLIN

Our objective is to learn robust and transferable universal facial representation from abundantly available non-annotated facial video data [78]. If we think holistically, face specific tasks involve two different aspects: a) facial appearance related attributes such as parts of the face (nose, eyes, lips, hair, etc.), facial shape and texture which mainly need spatial investigation; and b) facial action such as emotion, Facial Action Coding System (FACS), lip synchronization which requires temporal information. Thus, spatio-temporal modeling is highly desirable in order to learn strong, robust, and transferable representation. To this end, our proposed framework, MARLIN, adopts a facial region guided masking strategy which poses a challenging auxiliary reconstruction task for self supervised representation learning (See Fig. 2). To facilitate learning from masked auto-encoder, we mainly choose the YouTube Faces [78] dataset that uses web-crawled facial videos from YouTube having variation in terms of different real life conditions.

3.1. Facial Representation Learning

Preliminaries. MARLIN consists of an encoder (ϕ∈), decoder (ϕD) and discriminator (ϕΓ) with embedding parameters φϕ, φD, and φΓ, respectively. Given a training dataset D = {Xv}Nv=1 where N is the number of videos in the dataset and V ∈ RCV×H×W (C, T, H, W are channel, temporal depth, height and width of the raw video, respectively). From the raw input video V, we track and crop the facial regions [75] followed by random temporal sampling represented as v ∈ RCT×H×W (T, H, W are the modified temporal depth, height, and width of the derived video clip, respectively). The derived video clip v is further mapped to (k − n) visible and n masked tokens denoted as {Xv, n, Xm, n} by facial-region guided masking strategy (ϕeg) with a pre-defined masking ratio r = k. Here, e is the embedding dimension and k is the total number of tokens derived from v, i.e. k = T × H × W, given 3D cube tokens have dimension of t × h × w each. Thus, MARLIN injects facial region specific domain knowledge in the aforementioned token space to guide the representation learning via masking.

The visible tokens Xv, are mapped to the latent space z by the following mapping function Fϕ∈ : Xv → z. The latent space feature z is further fed to the decoder FϕD which reconstruct z to the n masked tokens Xm, by the following mapping FϕD : z → Xm. In the decoder, the corresponding visible and masked 3D cubes contain the flattened raw pixels denoted as e = Ctw. In brief given the visible tokens Xv, we reconstruct the masked tokens by the following function:

\[ X_m = F_{\phi_D} \circ F_{\phi_{eg}}(X_v) \] (1)

Reconstructing spatio-temporal facial patterns from raw pixels is quite challenging, we deploy a discriminator FϕΓ with the adversarial training for better synthesis.

3.2. Self-Supervised Representation Learning.

The self supervised pre-training strategy of MARLIN consists of three main components described below:

a) Facial-region Guided Tube Masking (Fasking). In order to capture spatio-temporal correspondence, we have deployed facial region specific tube masking strategy following [71]. We dynamically track and mask facial compo-
Thus, the decoder \( F \) reference to generate the masked counterpart of the face.

b) Masked Autoencoder. The masked spatio-temporal cubes look like deformable bending tubes, we termed it as Facial region-guided tube masking aka Fasking.

We begin with face parsing using FaceXZoo [75] library which divides facial regions into the following parts \{left-eye, right-eye, nose, mouth, hair, skin, background\} (Fig. 2 (b)). Among the facial regions, we prioritize the following set \( \mathbb{P} = \{\text{left-eye, right-eye, nose, mouth, hair}\} \) over skin and background to preserve face specific local and sparse features. In order to maintain pre-defined masking ratio \( r \), facial regions from the priority set \( \mathbb{P} \) are masked across frames first followed by \{background, skin\} masking. Thus, Fasking generates \( n \) masked and \( (k - n) \) visible tokens. Across all the frames of the input \( v \), we track specific facial regions from the pre-defined set to encode and reconstruct spatio-temporal changes to the model facial motion. The fasking strategy thus poses more challenges to the reconstruction while encoding subject specific appearance and fine-grained details.

b) Masked Autoencoder. After Fasking, \( (k - n) \) visible tokens are given input to the Encoder \( \mathcal{F}_{\phi_{v}} \) which maps the tokens to the latent space \( z \). The visible tokens serve as a reference to generate the masked counterpart of the face. Thus, the decoder \( \mathcal{F}_{\phi_{d}} \) maps the latent space \( z \) to the reconstructed masked tokens \( X_{m}' \). Please note that similar to VideoMAE [71], we adopt ViT [28] architecture as a backbone for MARLIN. A reconstruction loss \( \mathcal{L}_{\text{recon}} \) is imposed between masked cubes \( X_{m} \) and their reconstructed counterparts \( X_{m}' \) to guide the learning objective.

c) Adversarial Adaptation Strategy. To enhance the generation quality for rich representation learning, we incorporate adversarial adaptation on top of the masked auto-encoder backbone. According to the prior literature [27, 60], adversarial training enhances generation quality which in turn results in rich latent feature \( z \). The discriminator \( \mathcal{F}_{\phi_{d}} \) as shown in Fig. 2 is an MLP based network which imposes adversarial loss \( \mathcal{L}_{\text{adv}} \) between \( X_{m} \) and their reconstructed counterparts \( X_{m}' \).

3.3. Overall MARLIN Loss

Alg. 2 summarizes the training process for the MARLIN framework. MARLIN mainly imposes (a) Reconstruction Loss and (b) Adversarial Loss to facilitate the training.

(a) Reconstruction Loss. Given an input masked tokens \( \hat{X}_{m} \), the masked auto-encoder module reconstruct it back to \( X_{m}' \). To this end, we minimize mean squared error loss in the 3D token space to update the weights of the \( \mathcal{F}_{\phi_{v}} \circ \mathcal{F}_{\phi_{d}} \) branch. The loss is defined as

\[
\mathcal{L}_{\text{recon}} = \frac{1}{N} \sum_{i=1}^{N} \| X_{m}^{(i)} - X_{m}'^{(i)} \|_{2}^{2} \tag{2}
\]

where \( N \) is the total number of data in \( \mathcal{D} \), \( X_{m}^{(i)} \) and \( X_{m}'^{(i)} \) are the masked token and reconstruction of \( i \)-th data in \( \mathcal{D} \).
Adapt downstream label
Train Fasking (See Algo 1)
Fine-Tuning
Linear Probing
Train protocols, we adopt Linear Probing (LP) and Fine-Tuning (a self-supervised way. Following the standard evaluation transferable facial representation from the facial video in ϕ parameters.

Thus, the overall learning objective L is formulated as follows:

\[ L^{(d)}_{adv} = \frac{1}{Nn} \sum_{i=1}^{N} \left( \sum_{x_m' \in X_m^{(i)}} F_{\phi_T}(x_m') - \sum_{x_m \in X_m^{(i)}} F_{\phi_T}(x_m) \right) \]

(3)

\[ L^{(g)}_{adv} = -\frac{1}{Nn} \sum_{i=1}^{N} \sum_{x_m' \in X_m^{(i)}} F_{\phi_T}(x_m') \]

(4)

Thus, the overall learning objective L is formulated as follows, where \( \lambda_W \) is the weighting parameter:

\[ L^{(g)} = L^{recon} + \lambda_W L^{(g)}_{adv} \]

(5)

\[ L^{(d)} = L^{(d)}_{adv} \]

(6)

During MARLIN’s pre-training phase, \( L^{(d)} \) updates the parameters \( \phi_{d_{dis}} \) and \( L^{(g)} \) updates the parameters \( \phi_{e}, \phi_{d} \).

3.4. Downstream Adaptation

Our proposed MARLIN framework learns robust and transferable facial representation from the facial video in a self-supervised way. Following the standard evaluation protocols, we adopt Linear Probing (LP) and Fine-Tuning (FT) for downstream adaptation for different face relevant tasks (See Fig. 2 inference module). Given any task specific downstream dataset \( D_{down} = \{ v_j, y_j \}_{j=1}^{N} \), we deploy linear fully-connected (FC) layers with embedding parameters \( \theta \) to align the latent space to the downstream task specific label space on top of encoder module \( F_{\phi_e} \). For linear probing, we freeze the backbone network \( F_{\phi_e} \) and only update the \( F_{\phi_T} \). On the other hand for FT, we fine-tune the whole module i.e. \( F_{\phi_e} \circ F_{\phi_T} \). When MARLIN is used as a feature extractor for LP, it uses a sliding temporal window to extract features \( Z \) of the input face cropped video \( V \) as shown in Fig. 2 (c). The details of different downstream facial tasks are described below:

**Facial Attribute Recognition (FAR)** predicts the presence of appearance and action attributes such as gender, race, hair color, and emotion of a given face video. The problem of predicting facial attributes can be posed as a multi-label learning problem highly dependent on rich spatial encoding. For the downstream adaptation purpose, we use 28,532 train, 3,567 val, and 3,567 test data. Following the prior works [7, 25], we report average accuracy(↑), Area Under the Curve (AUC↑) over all attributes.

**Facial Expression Recognition (FER)** task encodes spatio-temporal facial muscle movement patterns to predict emotion (6-class) and sentiment (7-class and 2-class) of the concerned subject given a facial video. We evaluate the performance of MARLIN on CMU-MOSEI dataset [7] which is a conversational corpus having 16,726 train, 1,871 val, and 4,662 test data. Following the prior works [7, 25], we use overall accuracy(↑) as metrics.

**Deepfake Detection (DFD)** task predicts spatio-temporal facial forgery given a facial video from FF++(LQ) dataset [62]. For downstream adaptation, we use 3,600 train, 700 val, and 700 test sample videos from FF++(LQ) dataset [62]. Following prior literature [12, 58, 76], we use accuracy(↑) and AUC(↑) as the evaluation metrics.

**Lip Synchronization (LS)** is another line of research that require facial region specific spatio-temporal synchronization. This downstream adaptation further elaborates the adaptation capability of MARLIN for face generation tasks. For adaptation, we replace the facial encoder module in Wav2Lip [57] with MARLIN, and adjust the temporal window accordingly i.e. from 5 frames to T frames. For evaluation, we use the LRS2 [22] dataset having 45,838 train, 1,082 val, and 1,243 test videos. Following the prior literature [57, 74], we use Lip-Sync Error-Distance (LSE-D ↓), Lip-Sync Error-Confidence (LSE-C ↑) and Frechet Inception Distance (FID ↓) [38] as evaluation metrics.

4. Experiments and Results

We have comprehensively compared our method on different downstream adaptation tasks from quantitative (See Sec. 4.2) and qualitative (See Sec. 4.3 perspectives. Ad-
Table 2. Facial Attribute Recognition. Our proposed framework, MARLIN, trained on YTF [78] dataset and Linear Probed/Fine-Tuned on CelebV-HQ [85] benchmark dataset in terms of accuracy↑ and area under the curve↑. * shows supervised methods trained on the CelebV-HQ [85] dataset.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>R3D [72] *</td>
<td>92.34</td>
<td>0.9424</td>
<td>94.57</td>
<td>0.9173</td>
<td>93.45</td>
</tr>
<tr>
<td>MViTv1 [30] *</td>
<td>92.90</td>
<td>0.9452</td>
<td>95.13</td>
<td>0.9233</td>
<td>94.01</td>
</tr>
<tr>
<td>MViTv2 [49] *</td>
<td>92.77</td>
<td>0.954</td>
<td>95.15</td>
<td>0.9239</td>
<td>93.96</td>
</tr>
<tr>
<td>VideoMAE (FT) [71]*</td>
<td>92.91</td>
<td>0.9529</td>
<td>95.37</td>
<td>0.9284</td>
<td>94.14</td>
</tr>
<tr>
<td>MARLIN (LP)</td>
<td>91.90</td>
<td>0.9573</td>
<td>95.25</td>
<td>0.9278</td>
<td>93.57</td>
</tr>
<tr>
<td>MARLIN (FT)</td>
<td>93.90</td>
<td>0.9561</td>
<td>95.48</td>
<td>0.9406</td>
<td>94.69</td>
</tr>
</tbody>
</table>

Additionally, we have performed extensive ablation studies to provide justification for our design choices.

4.1. Experimental Protocols

Datasets. We evaluate the MARLIN framework on different facial analysis tasks described in Sec. 3.4. In brief, we use CelebV-HQ [85] for facial attribute and action prediction, CMU-MOSEI dataset [7] for conversational emotion and sentiment prediction, FF++(LQ) dataset [62] for deepfake detection and LRS2 [22] for lip synchronization.

Settings. For fair comparisons, we follow the dataset specific experimental protocols mentioned in the task specific prior literature [7, 22, 33, 50, 62, 84]. Other than traditional evaluation, we perform few shot adaptation strategy as well to show the robustness and transferability of MARLIN.

Implementation Details. We implemented the method on PyTorch [55] with Nvidia RTX A6000 GPU. First of all, given any temporal chunk of a facial video, consecutive frames are highly redundant. Therefore, to consider semantically meaningful frames having significant motion across frames, we adopt the minimum temporal stride value to be 2. Given an input video (having dimension 3 × 16 × 224 × 224), the cube embedding layer generates 8 × 14 × 14 3D tokens of dimension 2 × 16 × 16 to preserve spatio-temporal patterns. Using the Fasking strategy (See Algo. 1), MARLIN densely masks these tokens with a pre-defined masking ratio. Our empirical analysis suggests that MARLIN works favorably with a high masking ratio (90%). MARLIN’s objective is to generate the masked part from the sparse visible tokens. After Fasking, each token is mapped to the latent space embedding dimension of 768. From this latent embedding, the masked part is reconstructed in the 3D token space that can further be mapped to the original video. For fair comparison, we use ViT-B as the backbone encoder, although the impact of other ViT-variants are depicted in ablation study. The pre-training hyperparameters are as follows: the base learning rate is linearly scaled with respect to the overall batch size, $lr = \text{base learning rate} \times \text{batch size/256}$. For self-supervised pre-training, we use AdamW optimizer with base learning rate $1.5e-4$, momentum $\beta_1 = 0.9, \beta_2 = 0.95$ with a learning rate scheduler (cosine decay) [51]. For linear probing, we use Adam optimizer with $\beta_1 = 0.5, \beta_2 = 0.9$ and base learning rate $1e-4$, weight decay 0. For fine-tuning, we use Adam optimizer with $\beta_1 = 0.5, \beta_2 = 0.9$ and base learning rate $1e-4$ without any weight decay.

4.2. Quantitative Analysis

4.2.1. Comparison with SOTA Facial Analysis Tasks.

We compare the performance of MARLIN with different downstream facial analysis tasks following standard task specific evaluation protocols [7, 22, 33, 50, 62, 84].

Facial Attributes. In Tab. 2, we compare the LP and FT adaptation performance of MARLIN with the popular trans-

Table 3. Facial Expression and Sentiment Recognition. Downstream adaptation results on MOSEI dataset [7] for Emotion, sentiment (7-class), and sentiment (2-class). Our proposed method, MARLIN, outperforms visual modality based emotion prediction methods. Please note that SOTA for UMONS [25] and GMF [4] utilize three modalities and thus, not directly comparable. Here, YTF: YouTubeFace [78] and LAV represents linguistic, audio, and visual modality, respectively. * denotes supervised methods.

<table>
<thead>
<tr>
<th>Tasks</th>
<th>Pre-train</th>
<th>Method</th>
<th>Mod.</th>
<th>Acc.↑</th>
</tr>
</thead>
<tbody>
<tr>
<td>Emotion</td>
<td>–</td>
<td>MViTv1 [49]*</td>
<td>V</td>
<td>80.45</td>
</tr>
<tr>
<td></td>
<td>–</td>
<td>UMONS [25]*</td>
<td>V</td>
<td>80.68</td>
</tr>
<tr>
<td></td>
<td>–</td>
<td>GMF [4]*</td>
<td>LAV</td>
<td>81.14</td>
</tr>
<tr>
<td></td>
<td>–</td>
<td>VideoMAE [71]</td>
<td>V</td>
<td>80.39</td>
</tr>
<tr>
<td></td>
<td>–</td>
<td>MARLIN</td>
<td>V</td>
<td>80.60</td>
</tr>
<tr>
<td>Sentiment (7-Class)</td>
<td>–</td>
<td>MViTv1 [49]*</td>
<td>V</td>
<td>33.35</td>
</tr>
<tr>
<td>Sentiment (2-Class)</td>
<td>–</td>
<td>MARLIN</td>
<td>V</td>
<td>34.63</td>
</tr>
<tr>
<td></td>
<td>YTF [78]</td>
<td>VideoMAE [71]</td>
<td>V</td>
<td>33.78</td>
</tr>
<tr>
<td></td>
<td>YTF [78]</td>
<td>MARLIN</td>
<td>V</td>
<td>71.06</td>
</tr>
<tr>
<td></td>
<td>YTF [78]</td>
<td>CAE-LR [45]</td>
<td>V</td>
<td>72.96</td>
</tr>
<tr>
<td></td>
<td>YTF [78]</td>
<td>MARLIN</td>
<td>V</td>
<td>73.70</td>
</tr>
</tbody>
</table>

Table 4. Deepfake Detection. We compare the Fine-Tuning (FT) results on MARLIN for FaceForensic++ [62] dataset. * denotes supervised methods.

<table>
<thead>
<tr>
<th>Pre-train</th>
<th>Method</th>
<th>Acc.↑</th>
</tr>
</thead>
<tbody>
<tr>
<td>–</td>
<td>Steg.Features [32]*</td>
<td>55.98</td>
</tr>
<tr>
<td>–</td>
<td>LD-CNN [24]*</td>
<td>58.69</td>
</tr>
<tr>
<td>–</td>
<td>Constrained Conv. [8]*</td>
<td>66.84</td>
</tr>
<tr>
<td>–</td>
<td>CustomPooling CNN [61]*</td>
<td>61.18</td>
</tr>
<tr>
<td>–</td>
<td>Mesonet [23]*</td>
<td>70.47</td>
</tr>
<tr>
<td>–</td>
<td>Face X-ray [47]*</td>
<td>– 0.6160</td>
</tr>
<tr>
<td>–</td>
<td>Xception [21]*</td>
<td>86.86</td>
</tr>
<tr>
<td>–</td>
<td>F3-Net [58]*</td>
<td>93.02</td>
</tr>
<tr>
<td>–</td>
<td>P3D [59]*</td>
<td>– 0.6705</td>
</tr>
<tr>
<td>–</td>
<td>R3D [72]*</td>
<td>– 0.8772</td>
</tr>
<tr>
<td>–</td>
<td>I3D [15]*</td>
<td>– 0.9318</td>
</tr>
<tr>
<td>–</td>
<td>M2TR [76]*</td>
<td>– 0.9395</td>
</tr>
<tr>
<td>–</td>
<td>ST-M2TR [76]*</td>
<td>– 0.9531</td>
</tr>
<tr>
<td>YTF [78]</td>
<td>VideoMAE [71]</td>
<td>87.57</td>
</tr>
<tr>
<td>YTF [78]</td>
<td>MARLIN</td>
<td>89.43</td>
</tr>
</tbody>
</table>

0.9305
former (i.e. MViT-v1 [30] and MViT-v2 [49]) and CNNs (i.e. R3D [72]) on CelebV-HQ [85] dataset. From the table, it is observed that MARLIN’s FT version outperforms supervised MViT-v2 [49] transformer architecture by 1.13% (92.77% → 93.90%) for appearance attributes and 0.33% (95.15% → 95.48%) for action attributes. Similar patterns are also been observed with the R3D CNN module as well. We attribute MARLIN’s performance gain to the pre-training strategy that encodes generic, robust, and transferable features from any input facial video.

**Emotion and Sentiment.** In Tab. 3, we similarly compare the LP and FT adaptation performance of conversational emotion and sentiment in terms of accuracy(↑) and AUC(↑) on CMU-MOSEI [83] dataset. Please note that the MARLIN is a visual modality only encoder. The results suggest that MARLIN performs competitively with SOTA methods [25, 45, 49], especially it outperforms unsupervised SOTA CAE-LR [45] by 2.64% (71.06% → 73.70%) on 2-class sentiment task. For emotion and 7-class sentiment as well, it outperforms supervised benchmarks [49] marginally. These results also indicate that MARLIN learns highly generic, robust, and transferable feature representation from pre-training.

**DeepFake Detection.** In Tab. 4, we compare the performance of video manipulation on FaceForensics++ [62] dataset and report results in terms of video-level accuracy(↑) and AUC(↑). The results indicate that MARLIN performs favorably against the supervised SOTA methods [2, 8, 15, 21, 24, 32, 47, 59, 61, 72]. This is the first SSL work that uses only spatio-temporal visual information anomaly to detect video manipulation. Unlike F3-Net, which uses frequency aware pattern over the temporal dimension to detect forgeries in a supervised fashion. Whereas MARLIN irrespective of frequency pattern learns facial representation and can detect anomalies from the spatio-temporal signal.

**Lip Synchronization.** For a fair comparison, we adopt the following experimental setups: 1) Wav2Lip+ViT: To compare the contribution of ViT architecture [28] wrt SOTA CNNs and MARLIN where the weights of ViT is trained from scratch on LRS2 [22] dataset. 2) Wav2Lip+ViT+VideoMAE: To compare the contribution of vanilla VideoMAE with ViT backbone pre-trained on YTF [78] dataset. 2) Wav2Lip+ViT+MARLIN: To compare the contribution of MARLIN pre-trained on YTF [78] with SOTA [57, 66, 74] and different design aspects. The experimental results are depicted in Tab. 5 with LSE-D↓, LSE-C↑ and FID↓ as evaluation metrics following standard protocol [38, 57, 66, 74]. The improvement of lip sync score (LSE-D↓: 7.521 → 7.127; FID↓: 4.887 → 3.452) indicates that MARLIN learns rich spatio-temporal patterns which are transferable and robust. It is also interesting to observe that MARLIN is adaptive to very fine grained features specific to the face as well.

**4.2.2. Few-Shot Adaptation.**
Few shot adaptation has recently gained attention due to its adaptation capability with very low data regime [9, 65, 84, 86]. Following the standard evaluation protocol [9, 65, 84, 86], we also investigate the adaptation capability of MARLIN. Given any downstream dataset, we use limited train set labels to align the output manifold while keeping the test set fixed via LP (MOSEI, CelebV-HQ) and FT (FF+) strategy. From Tab. 6, a slight drop in performance is observed across different tasks which further demonstrates that MARLIN learns generic, transferable, and adaptive information.

**4.2.3. Ablation Studies.**
We have performed extensive ablation studies to show the contribution of different modules, encoder architectures, and masking strategies towards overall MARLIN framework. Fasking: Facial Guided Masking, AT: Adversarial Training
effectiveness of each component.

1) Masking ratio. We use different masking ratios in the range [0.05 - 0.95] and repeat the pre-training followed by LP on CMU-MOSEI [83] dataset. From Fig. 3, we see that ∼ 90% masking ratio is optimal for the MARLIN. With a less masking ratio (i.e. ≤ 0.5), more information is available for the reconstruction task which degrades the feature quality. Similarly, beyond ∼ 90%, the reconstruction task becomes more challenging, leading to a performance drop. With the empirical evidence, we set the masking ratio to be ∼ 90% throughout all of our experiments. 2) Masking strategies. We further compare the proposed Fasking strategy with existing masking strategies [31,71] i.e. Frame, Random and Tube-Masking. The empirical results in Tab. 7 demonstrate that Fasking is better.

3) Different modules. We progressively integrate each module and observe its impact on downstream performance on CMU-MOSEI [83] and FF++ [62] while keeping other components fixed. From Tab. 7, we see that the addition of Fasking and Adversarial Training (AT) improves the performance, reflecting the importance of each component. 4) Encoder architectures. To investigate the impact of the backbone encoder architectures, and compare ViT-S, ViT-B, and ViT-L (See Tab. 7). We observe that the larger model size enhances the performance. For fair comparison, we use a ViT-B encoder.

4.3 Qualitative Aspects

In order to understand the effectiveness of the learned features, we further conducted following qualitative analysis.

1) Facial Attributes. We visualize the important regions that MARLIN focused on using Gradient-weighted Class Activation Mapping (Grad-CAM) [64]. In Fig. 4 top, the heat-map results are based on LP on top of MARLIN’s feature on CelebV-HQ [85] dataset (appearance task) and it indicates that MARLIN focus on facial attributes such as hair, spectacle, hat, etc. 2) Lip Synchronization. In Fig. 4 bottom, we presents the generation results for lower part of faces which is a challenging task. The top, middle and bottom rows show ground truth, vanilla Wav2Lip [57]'s output and MARLIN’s output along with the closeup looks, respectively. Here, Wav2lip’s CNN encoder failed to locate the lip region (as shown in the Wav2lip row of Fig. 4 highlighted in red) whereas MARLIN despite pre-trained on fasking strategy is adaptive enough to generate more accu-
References


[24] Davide Cozzolino, Giovanni Poggi, and Luisa Verdoliva. Recasting Residual-based Local Descriptors as Convolutional Neural Networks: an Application to Image Forgery Detecc-


